

AD-A184489

NAVAL OCEAN SYSTEMS CENTER, SAN DIEGO, CA  
INTERFACING DISTRIBUTED OPERATING SYSTEMS  
BY: A ECHENIQUE

1 OF 1  
NOSC TD 1120  
UNCLASSIFIED  
JUL 87

END  
DATE  
FILMED

**NOSC**  
NAVAL OCEAN SYSTEMS CENTER San Diego, California 92152-5000

**Technical Document 1120**  
July 1987

# Interfacing Distributed Operating Systems

A. Echenique  
BBN Laboratories Incorporated



Approved for public release;  
distribution is unlimited

The views and conclusions contained in this report are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Naval Ocean Systems Center or the U S government

# NAVAL OCEAN SYSTEMS CENTER

San Diego, California 92152-5000

---

E. G. SCHWEIZER, CAPT, USN  
Commander

R. M. HILLYER  
Technical Director

## ADMINISTRATIVE INFORMATION

This report was prepared by BBN Laboratories Incorporated under contract N66001-86-D-0084 for Code 443 of the Naval Ocean Systems Center, San Diego, CA 92152-5000.

Released by  
D.C. McCall, Head  
C<sup>2</sup> Information Processing  
and Display Technology  
Branch

Under authority of  
W.T. Rasmussen, Head  
Advanced C<sup>2</sup> Technologies  
Division

# REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION <b>UNCLASSIFIED</b>			1b RESTRICTIVE MARKINGS		
2a SECURITY CLASSIFICATION AUTHORITY			3 DISTRIBUTION/AVAILABILITY OF REPORT  Approved for public release; distribution is unlimited.		
2b DECLASSIFICATION/DOWNGRADING SCHEDULE					
4 PERFORMING ORGANIZATION REPORT NUMBER(S)  Report Number 6375			5 MONITORING ORGANIZATION REPORT NUMBER(S)  NOSC TD 1120		
6a NAME OF PERFORMING ORGANIZATION  BBN Laboratories, Inc.		6b OFFICE SYMBOL (If applicable)	7a NAME OF MONITORING ORGANIZATION  Naval Ocean Systems Center		
6c ADDRESS (City, State and ZIP Code)  10 Moulton Street Cambridge, MA 02238			7b ADDRESS (City, State and ZIP Code)  San Diego, CA 92152-5000		
8a NAME OF FUNDING/SPONSORING ORGANIZATION  Office of Naval Technology		8b OFFICE SYMBOL (If applicable)  ONT	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER  N66001-86-D-0084		
8c ADDRESS (City, State and ZIP Code)  800 N. Quincy Arlington, VA 22217-5000			10 SOURCE OF FUNDING NUMBERS		
			PROGRAM ELEMENT NO  62721N	PROJECT NO  CDB3	TASK NO  AGENCY ACCESSION NO DN888 630
11 TITLE (Include Security Classification)  Interfacing Distributed Operating Systems					
12 PERSONAL AUTHOR(S)  A. Echenique					
13a TYPE OF REPORT  Final		13b TIME COVERED FROM <u>Oct 86</u> TO <u>Oct 86</u>		14 DATE OF REPORT (Year, Month, Day)  July 1987	
15 PAGE COUNT  26					
16 SUPPLEMENTARY NOTATION					
17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB GROUP			
			Cronus cluster		
			Link 11 Afloat Correlation System (ACS)		
19 ABSTRACT (Continue on reverse if necessary and identify by block number)  BBN has developed a distributed operating system, Cronus, which functions in the context of a heterogeneous internetwork system architecture. Cronus is intended to introduce coherence and uniformity to a set of otherwise independent and disjoint computer systems. Cronus also provides a development, methodology, tools and mechanisms to support distributed application development and support for functional properties such as scalability, resource management, and survivability.					
20 DISTRIBUTION AVAILABILITY OF ABSTRACT <input type="checkbox"/> UNCLASSIFIED UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS			21 ABSTRACT SECURITY CLASSIFICATION <b>UNCLASSIFIED</b>		
22a NAME OF RESPONSIBLE INDIVIDUAL  D. Teasdale			22b TELEPHONE (Include Area Code)  (619)225-7485		22c OFFICE SYMBOL  Code 443

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

DD FORM 1473, 84 JAN

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

## Introduction

Several significant exploratory and advanced development programs at NOSC are investigating the use of distributed processing methods to improve fleet command, control and targeting systems. Tactical anti-submarine warfare (ASW) information processing systems, for example, will require significantly increased connectivity with computer systems on both their own platform as well as other platforms. In the upgraded ASW systems, new data sources will be provided from systems such as ACS (Afloat Correlation System) to improve the surface scene description and to allow better coordination of the various platforms. This need for increased connectivity among various systems may be well served by the application of distributed operating system technology. In this role, distributed operating system technology can provide an efficient and effective means for the integration of various and diverse command and control data processing tasks.

BBN has developed a distributed operating system, Cronus, which functions in the context of a heterogeneous internetwork system architecture. Cronus is intended to introduce coherence and uniformity to a set of otherwise independent and disjoint computer systems. Cronus also provides a development methodology, tools and mechanisms to support distributed application development and support for functional properties such as scalability, resource management, and survivability.

The Cronus architecture provides a flexible environment for interconnecting hosts so that facilities available on one host may be conveniently used from other hosts. Varying degrees of host integration are supported. A complete host integration supports a well

## **Report No. 6375 Interfacing Distributed Operating Systems**

developed interprocess communication mechanism. This interprocess communication mechanism provides all the functions and services needed to simultaneously support a variety of distributed processing tasks. More limited host integration schemes focus on the support of particular distributed processing functions such as remote login.

The need to provide greater connectivity among various Navy command and control systems can be served by the development of specialized Cronus interfaces. In general, these specialized Cronus interfaces, referred to as Access Systems, are examples of limited host integration schemes which support the task of acquisition, translation and transferral of data from an external, non-Cronus information system to a variety of clients operating in a Cronus cluster.

A Cronus testbed featuring an Access System to an external information processing system can provide an additional integrating factor for Navy applications and data sources. Such a testbed will allow a diverse set of applications operating on a heterogenous set of hosts to process external data in a variety of ways not otherwise possible. Of particular interest for such applications, is the development of an interface between a Cronus system and the Naval Tactical Data System (NTDS/Link-11 system). A system to provide access to Link-11 tactical data via Cronus would allow development and testing of Cronus applications in the field of command and control and the presentation of meaningful demonstrations with these applications in a realistic environment.

This report explores issues related to the functional design of a Cronus/Link-11 access system and the selection of a suitable hardware base for its implementation. This document will also develop a framework for the integration of an access system into a NCSC Cronus testbed and will highlight applications which are made possible by such an access system. Two alternative architectures for such a Cronus/Link-11 Access System will be presented. These architectures have been selected and developed on the basis of

## **Report No. 6375 Interfacing Distributed Operating Systems**

the hardware available to implement these systems. From these two architectures, three possible implementation schemes are presented. The three implementation schemes are examined and evaluated in terms of their cost, functionality and performance. This document does not address the numerous detailed technical issues related to the actual implementation of a Cronus/Link-11 access system. Instead, the development of these technical details is best withheld until selection is made regarding possible Access System architectures and implementations.

### **§1.0 NTDS Interface Availability**

Suitable alternatives for the implementation of a Cronus/Link-11 Access System are ultimately limited by the availability of hardware devices which can form electrical interfaces between the NTDS/Link-11 system and other systems which are able to support Cronus functionality. From this starting point, investigation into a Cronus/Link-11 Access System must begin with the evaluation of the hardware and software systems which can provide this basic interface function.

Unfortunately, there are only two interfaces currently available which can support the Cronus/Link-11 Access System. The first is an interface device which interconnects the NTDS/Link-11 electrical protocol and the IBM PC I/O bus. The other device interconnects the NTDS/Link-11 electrical protocol with the DEC LSI-11/23 Q-bus. The following sections highlight relevant features of these devices and the software to support them. A third alternative, that of a custom hardware device, will also be briefly discussed.

#### **§1.1 PC Hardware Alternative**

There are at least three interfaces available for interconnecting an IBM PC and the NTDS/Link-11 data system. These devices are manufactured by Rockwell International American Systems Corporation and Sabtech Industries. The Rockwell unit is not a

commercial product and thus information about the device is not available. The Sabtech Industries unit is in use at NOSC and was recommended over the American Systems unit. The Sabtech unit, Model NT1632FS, completely implements NTDS MIL STD 1397A and can be configured to run in either NTDS 'slow' mode or NTDS 'fast' mode (16 or 32 bit operation). Two software integration tools are also available for this device. The first is a stand-alone NTDS message generator which provides capability to quickly perform loopback tests and perform general system tests. The second software tool is a set of assembly language routines which provide high level language access to the interface. The existence of these tools allows for rapid development of new PC system software to interface to NTDS/Link-11.

The interface is moderately priced: total cost is about \$3700, which consists of \$2900 for the board, \$600 for the software and an additional \$200 for cables, manuals and loopback plugs.

### **§1.2 Q-bus Hardware Alternative**

A number of years ago, Advanced Computer Communications, Inc developed an LSI-11/NTDS interface (called the IF11Q/NTDS). This device is DEC Q-bus compatible and has been used in a number of Navy applications. In the interest of increasing the applicability of this device, ACC developed hardware and software modifications for the interface to allow compatibility between the  $\mu$ VAX Q-bus-II and the NTDS interface. This device has limited availability and is expensive: approximately \$13000 for the board and associated cables and software. An accurate breakdown of this price was not available since the IF11Q/NTDS product is only available on request for quotation basis. Pricing and lead times will be determined at that time based on ACC staffing requirements and engineering priorities. Interface integration problems are likely to occur due to the limited application of this interface to the VAX environment and the solution of these problems will likely increase the cost of this interface.

### **§1.3 Custom Hardware Alternative**

A third alternative for the interfacing to Link-11 is the development of new hardware. Currently there is no device available to interconnect NTDS and other common microcomputer standards such as the Multibus architecture used in the SUN 2 workstation, or VME bus architecture used in the SUN 3 workstation. Since both of these machines support Cronus, development of such an interface may prove useful for future flexibility and expansion of a NOSC Cronus testbed.

### **§2 Access System Architecture**

Before presenting descriptions of the Access System architectures made possible by the NTDS interfaces described in the previous section, it is useful to examine the basic elements of a generalized Access System. A Cronus Access System consists of three principal functional elements:

- **External Information System Interface and Interface Driver/Server:** This element allows hardware-level access between the external system and a Cronus host and provides high-level I/O and low-level protocol services necessary to support this access.
- **Translation Module:** This element performs the necessary conversions between external message formats and Cronus message formats to allow data exchange between the two systems.
- **Cronus External Data Manager:** The External Data Manager is a Cronus process or set of processes responsible for control of communications with the external system and the manipulation of external data within the Cronus environment. The External Data manager provides access to this external data for other Cronus processes and applications. The Manager may also provide and manage bulk storage of data to be exchanged between other Cronus Managers and applications and the External system.

In general, models of how to implement a Cronus access system can be constructed by mapping the above functional elements onto suitable hardware components. The following sections describe two architectures for the implementation of a NTDS/Link-11 to Cronus access system based on the hardware options presented in §1.1 and §1.2

## **§2.1 Gateway Model**

The gateway model of a Cronus access system is based on the division of responsibility between two physical systems: a gateway to perform protocol translation and a Cronus Link-11 Manager to provide access to Link-11 data to other Cronus processes and applications (see Figure 1). The model makes use of the following hardware components:

- An IBM-PC-AT or equivalent
- An IBM-PC/NTDS interface
- An IBM-PC/Ethernet interface
- A Cronus host with an Ethernet connection

In this model, the PC implements the following access system components:

- A Link-11 Server to manage the Link-11 interface and support the protocols associated with the Link-11 network system.
- A TCP/IP Module to manage the Ethernet interface and provide standard DoD protocol support for communication with a Link-11 Manager.
- A Translation Module to provide two-way translation of Link-11 messages and TCP/IP messages.
- A Connection Server to manage TCP/IP connections with other Cronus hosts and provide access to control facilities for the PC.

The Cronus host in this model, referred to as the Access Machine, in addition to running the standard Cronus kernel, runs a Cronus Link-11 Manager which acts as the External Data Manager. This manager is responsible for the establishment and control of TCP/IP links to the PC, for the transfer of data to and from the PC and for the provision of data manipulation mechanisms to allow convenient access to Link-11 data by other Cronus managers and applications. The Link-11 Manager may also be responsible for the management and storage of large sets of data to be exchanged between the two systems. The degree of functionality associated with this data storage capacity will be determined by the needs of applications which must access and process this stored data.

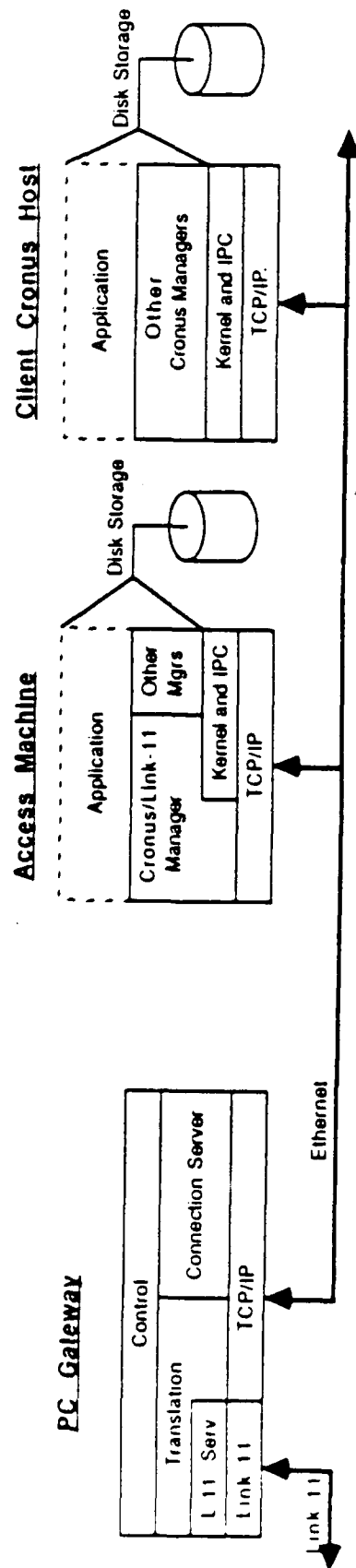


Figure 1. Gateway Model. In this model, the PC functions as a specialized internetwork gateway between Link-11 and TCP/IP. There is no Cronus functionality implemented on the PC. Cronus management of Link-11 data is accomplished on a separate machine, referred to as an Access Machine.

It is important to note that the PC in this model implements no Cronus functionality. In this respect, the PC functions strictly as a specialized internetwork gateway which performs message format translations and protocol translations between the Link-11 system and a generic TCP/IP environment. It is for this reason that this model is referred to as the "Gateway Model".

The Gateway Model presents numerous advantages for development. At a fundamental level, both the PC and the PC/Link-11 interface are inexpensive and readily available. Significant portions of the software to implement the PC segment of the access system are commercially available as development tools. New software for the PC would be limited to the control and translation functions and to code to integrate commercially available packages. Also, since the Gateway Model does not require that any Cronus functionality be implemented on the PC, all new Cronus code (the Link-11 Manager, etc.) would be implemented on 'familiar' systems, eg,  $\mu$ VAX or SUN.

Following is a summary of costs and issues associated with the development of a PC-based Gateway Model access system:

- Machine costs are low, approximately \$9700
- Good option for short term implementation, ie, there is only a moderate amount of software of known complexity that need be developed
- Inexpensive approach for both prototype stages involving few units and later deployment stages involving greater number of units
- Exhibits highly specialized functionality, ie, there is no actual Cronus features are supported
- Due to split of functionality across two separate machines, there may be limitations in performance, security and reliability.

## **§2.2 Access Machine Model**

The hardware required to implement the Access Machine Model of a Cronus/Link-11 access system is variable. Both the IBM-PC-AT with an NTDS/Link-11 interface and the

$\mu$ VAX with an NTDS/Link-11 interface could fulfill the basic requirements for implementation of this model. Selection between these alternative configurations is dependent upon actual and well defined needs for functionality and performance.

In the Access Machine model, the Access Machine implements the following access system components (see Figure 2):

- A Link-11 Server to manage the Link-11 interface and support the protocols associated with the Link-11 network system.
- A Translation Module to provide two-way translation of Link-11 messages and Cronus Link-11 Manager messages.
- A Cronus Link-11 Manager to provide Link-11 data manipulation mechanisms for other Cronus processes and applications and provide access to the Link-11 Server.
- An Cronus Interprocess Communications Module to support communications with other Cronus processes and applications.
- A TCP/IP Server to manage the Ethernet interface and provide standard protocol support for the Cronus Interprocess Communications module.

*The Access Machine model differs in architecture from the Gateway Model in that the Cronus Link-11 Manager is resident on the same machine that directly supports the Link-11 interface and Link-11 server. In the Gateway Model, the Link-11 Manager is not present on the Link-11 interface machine, but resides on some other Cronus host.*

As mentioned at the beginning of this section, the Access Machine Model may be implemented using either a  $\mu$ VAX or a PC. The  $\mu$ VAX and the PC present significantly different architectures and performance characteristics. Although the PC may be capable of supporting various segments of Cronus functionality, its lesser processing capacity may not leave sufficient reserve to provide an adequate degree of flexibility and extensibility for future applications. Conversely, the additional processing power provided by a  $\mu$ VAX will make possible a significantly more versatile foundation for implementation of an access system at additional cost.

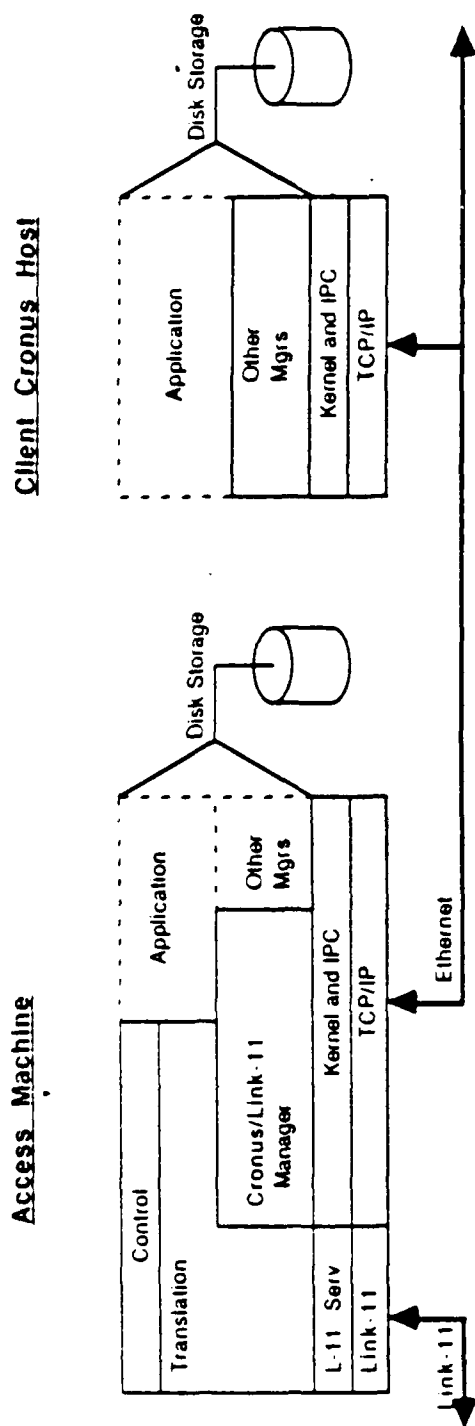


Figure 2. Access Machine Model. In this model, the actual machine which interconnects Link-11 and the Ethernet also implements the Cronus Manager to maintain and provide access to Link-11 data for other Cronus processes and applications. The access machine itself is may be implemented on either a PC or a larger machine such as a MicroVAX.

## **Report No. 6375 Interfacing Distributed Operating Systems**

There are two approaches to use of a PC for an access machine. The first involves porting a Cronus kernel and other software components to the PC through the use of commercially available PC UNIX operating system. The second approach involves development of a new multitasking system for the PC which is specially tailored to the needs of Cronus. Both approaches have their merits and costs.

Using a commercial version of PC UNIX will allow for a relatively rapid and easy implementation of Cronus functionality on the PC. There will likely have to be modifications to many aspects of the Cronus kernel and associated software, but the amount of effort to accomplish this will be relatively small. A certain amount of effort will also have to be expended initially to analyze carefully various UNIX systems available for the PC and their impact on a port of Cronus. A limitation that use of PC UNIX may present will be with respect to performance. It is possible that Cronus is too large an application to run efficiently on a machine such as the PC supporting UNIX. Quantitative estimates of this limitation are difficult at this stage of investigation.

The second approach to a PC-based access machine is via development of a custom multitasking system for the PC to support Cronus directly. The principle advantage to this approach is in performance. A custom multitasking system tailored to the needs of Cronus can be optimized to provide high performance. This may be a significant issue for a relatively small machine such as the PC, but will ultimately depend on performance needs of Cronus applications using the access system. Support for multitasking has been previously implemented on other machines for Cronus, but the amount of effort to achieve this for the PC is still undetermined due to implementation complexities caused by the PC's segmented memory architecture. In addition to the development needed to support multitasking, new device drivers for the multitasking environment must be written and tested. Once a complete multitasking environment has been established, a port of the Cronus kernel and other system software must be

performed. Following this, other software to support access machine functionality must be developed. The principal software component to be developed is a Link-11 manager. This manager will support NTDS/Link-11 protocol service, Cronus/Link-11 format translations, Link-11 data storage, and access mechanisms for other Cronus managers and applications.

Another factor which will affect the applicability of the PC to the Access Machine Model is the need for bulk storage of data between Cronus and the Link-11 system. The PC may present limitations in the quantity of bulk storage and the rate of access provided for applications which require such functionality regardless of what approach to PC implementation is used. Quantitative analyses of such limitations are difficult without detailed characteristics of the bulk data access needs for typical Navy tactical applications.

There are several significant advantages to an Access Machine model implemented on a  $\mu$ VAX. The first is that the  $\mu$ VAX makes available significantly increased processing power. This extra processing capacity will allow for a more sophisticated Link-11 Manager which will be capable of supporting complex access schemes for other managers and applications. A second advantage of use of a  $\mu$ VAX to support Link-11, is that efforts to implement this system would largely involve the development of new features for a familiar system which already supports Cronus, and in this sense provide additional leverage from previous development efforts. New code for a  $\mu$ VAX access machine would involve the development of system level code to support the NTDS/Link-11 interface itself, and the development of a Link-11 manager to support access to the Link-11 data for other Cronus managers and applications.

An additional advantage of a  $\mu$ VAX implementation is that capacities for bulk storage of Link-11 data are significantly greater than those for the PC. The  $\mu$ VAX provides capability for very large amounts of bulk storage and has the processing speed to allow

## Report No. 6375 Interfacing Distributed Operating Systems

rapid and flexible access to such storage. Because of this, the  $\mu$ VAX will allow much greater flexibility in providing bulk storage facilities for tactical applications than will a PC-based access machine.

A summary of the costs and issues related to a PC implementation of an Access Machine is as follows:

- PC and Link-11 interface are relatively inexpensive.
- Task can involve either short term development or longer term development depending on needs for performance. PC-UNIX version is relatively easy to implement, but may offer limited performance; a custom multitasking system will be a significant effort, but will provide a high-performance, optimized system.
- All development and operation occurs on a single type of machine; this may be an advantage in terms of reliability and ease of development.
- May be expensive alternative if only a few access systems are to be procured; cost effectiveness of approach will appear with larger systems.
- May provide limited bulk storage facilities.

A summary of the issues and costs related to a  $\mu$ VAX implementation of an Access Machine is as follows:

- Task consists of relatively short term development. Cronus kernel and other software already exists; new code limited to Link-11 server and driver and Link-11 Manager.
- $\mu$ VAX and Link-11 interface expensive, but a single system may be cost competitive with PC-based system if  $\mu$ VAX is already available for use; system very expensive for deployment applications involving many units.
- Development occurs on a single machine with an existing well-developed and tested Cronus environment; this will lead to a shorter development term.
- Can provide significant facilities for bulk storage.

### §3 Applications for a Cronus/Link-11 Access System

An immediate application for a Cronus Link-11 Access System is to provide an entry point for actual tactical data and allow for real-time processing of this data. In this role, the access system serves to aid both in the presentation of meaningful NCSC demonstrations, and also in the support of additional development of distributed applications.

A testbed which includes an access system will readily support varied efforts in the development of distributed applications for the Navy tactical environment. Since the access system performs all the translation functions necessary to allow heterogeneous Cronus clients to freely use Link-11 data, applications which implement tactical databases, tactical graphics displays and demonstrate other application integration features of the Cronus distributed operating system can be readily assembled. A variety of hosts can function both cooperatively and individually to process Link-11 data and thus integrate various discrete data processing tasks under a single system. For example, existing target correlation, target track prediction and target prioritization programs can be made to function cooperatively through simultaneous use of the Link-11 Access System. The Access System can also be made to provide long-term storage of Link-11 data sets for later retrieval and processing and for use in testing new applications. See Figure 3.

Demonstration of such applications can also be made extremely effective through use of the Cronus Monitoring and Control system. This system provides passive monitoring of the status and performance of various system components and controls various parameters affecting the management of system resources. Displays of both instantaneous and aggregate system data can be displayed in graphical format. This feature allows for simple and effective demonstrations of Cronus interoperability features.

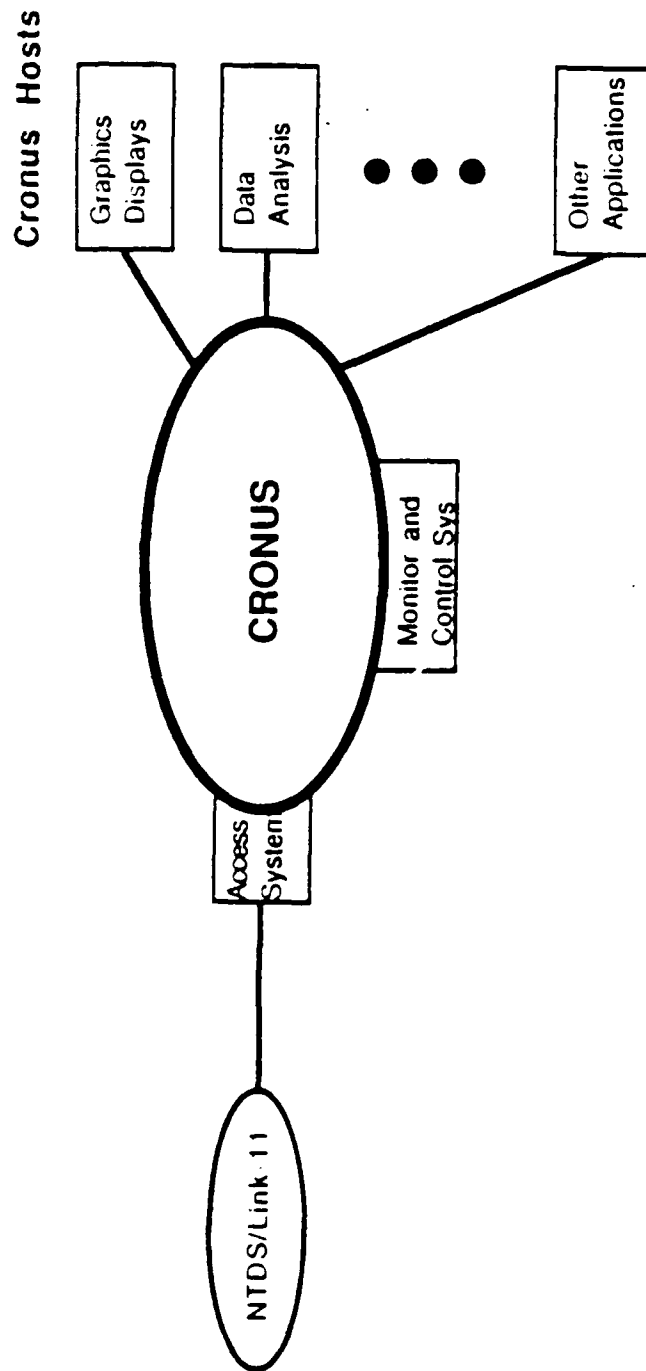


Figure 3. A Cronus testbed with a Link-11 Access System can provide access to tactical data for a variety of applications implemented on a heterogeneous group of hosts.

Cronus can also provide the basic features needed to support significant development activity in many aspects of survivable systems. Two fundamental elements of survivable system design are readily supported by Cronus:

- Data Survivability. Cronus supports replication of data objects. This low level replication feature in cooperation with resource monitoring and control, can form the basis for the development of databases capable of surviving component failures.
- System Configuration Management and Control. An essential requirement for a survivable system is the ability to monitor the status of system components and perform resource management based on this status information. The Cronus Monitoring and Control System can form the basis of efforts to develop system features which will detect component failures and control system configuration and operation in hostile environments.

Cronus applications can be developed which use these replication and monitoring and control facilities to implement survivable systems. However, to completely apply Cronus facilities for survivability, a testbed which implements some degree of survivability at the hardware subsystem level is required. Development of a Cronus/Link-11 Access System provides an opportunity to construct a rich yet inexpensive testbed which includes hardware subsystem survivability. Such a testbed can be cost effective and easily assembled because survivability is achieved through the replication of inexpensive and readily available subsystems.

A possible architecture for such a testbed would consist of two or more gateway-style PC-based access systems linked into the NTDS/Link-11 system. Each PC supporting the interface to Link-11 would reside on different Ethernets. One or more  $\mu$ VAX or SUN client machines would also be present on each of these Ethernets. In addition, at least one of the client machines on each Ethernet would be dually-homed to the other Ethernet. This provides fully redundant Ethernet connectivity between each client or group of client machines and each access point to Link-11. See Figure 4

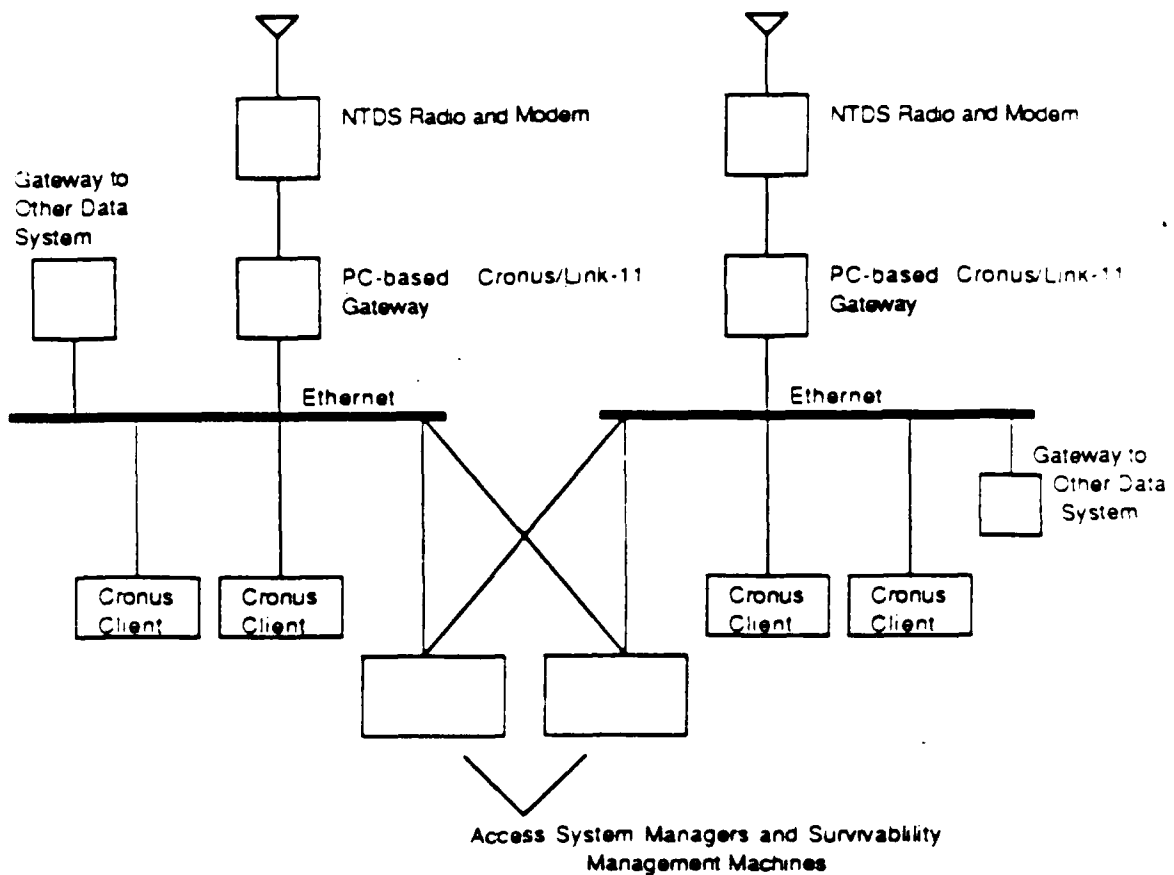


Figure 4. Survivable Cronus Testbed. This testbed implements the subsystem redundancy necessary for the development, testing and demonstration of Cronus applications oriented towards survivability. Note that the Access System Managers are capable of supporting different types of Access Systems. Also note that the Managers are installed on the dually-homed hosts to allow for Manager redundancy. Also note that this interconnection scheme is not limited to supporting just two-way redundancy. By grouping together subsystems, this testbed can be expanded to support even larger experiments and demonstrations.

This configuration supplies redundancies in several key areas. First, there are redundant links to the external data source, Link-11. Second, should there be either an access system failure or any client host failure, data can still be processed remotely from the other set of client hosts via the dually-homed host. Third, there is complete system redundancy in the case of an Ethernet failure.

In this testbed, Cronus can provide the necessary control and replication functions to implement survivability. Applications which utilize the Cronus Monitoring and Control system can be developed to perform system management and configuration control. These applications must monitor the performance of various elements of the testbed and command and control system reconfigurations in the event of subsystem failure. Through the use of other Cronus distributed processing facilities such support for object replication, other applications can be developed on this testbed which implement survivable databases. In this environment Cronus, can support work on survivability problems such as the resolution of discrepancies among replicated data records and recovery from database support subsystem failures.

#### §4 Conclusions

In this document, it has been shown that a Cronus/Link-11 Access System can provide an additional degree of integration for the various and diverse command and control data processing tasks in the Navy tactical environment. Three alternative schemes for the implementation of such an access system have been presented which reflect practical solutions to the access system problem. The merits and costs of these implementations are summarized below:

##### PC-Based Gateway Model

Basic Hardware Cost:

Low interface and machine under  
\$11K

## Report No. 6375 Interfacing Distributed Operating Systems

Development Time:	Short: no Cronus on PC; Link-11 Manager will be built for 'mature' Cronus host.
Functionality:	PC supports no Cronus functionality; split in functionality between two machines may present limitations in security, performance and reliability. PC cannot directly support storage but storage function can be applied to Cronus host supporting Link-11 manager.
Cost Effectiveness:	
Small Quantities:	High.
Larger Quantities:	High.

### PC-Based Access Machine Model

Basic Hardware Cost:	Low: interface and machine under \$11K.
Development Time:	Short term or longer term depending on needs for performance: PC UNIX can support rapid implementation will possibly limited performance; a custom multitasking environment can provide higher performance and optimization.
Functionality:	Since system is built on a single machine which communicates with other Cronus hosts via IPC, reliability and security will be high. Performance will depend on implementation. Storage capacity will be limited.
Cost Effectiveness:	
Small Quantities:	High if PC UNIX is used; lower if custom multitasking system is developed.
Larger Quantities:	Higher.

### uVAX-Based Access Machine Model

Basic Hardware Cost:	High: especially if processor must be newly procured.
----------------------	---

All three implementation schemes provide the necessary integration between NTDS/Link-11 and a Cronus testbed and provide broad opportunities for the development and demonstration of key aspects of distributed processing technology and for future growth and research. Selection among the schemes will be based on needs for functionality and testbed flexibility and the resources available to support the access system task.

**END  
DATE  
FILMED**

8-12-87